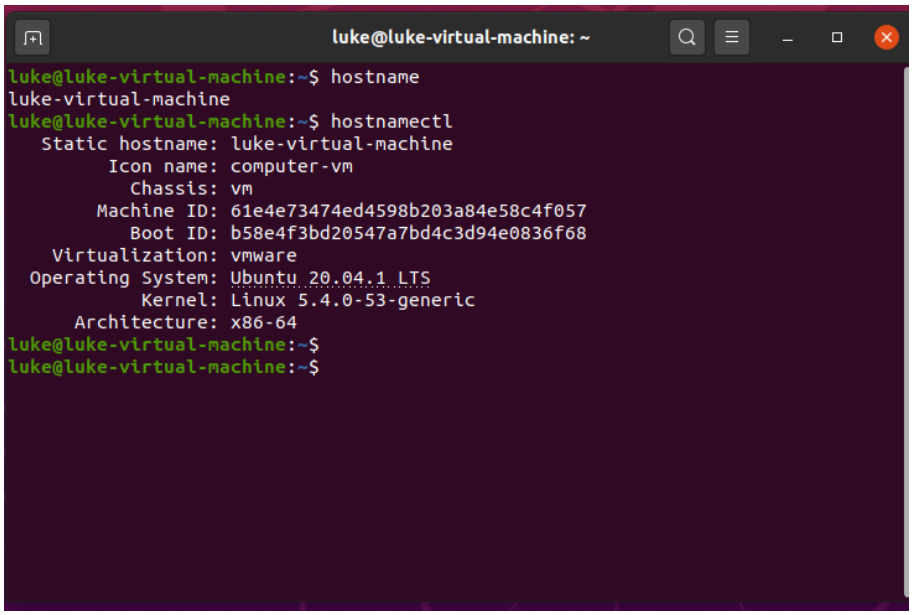
Problem 0. Briefly and clearly report what you have done for Homework Assignment #4.

Problem1

Sudo fdisk –l >after.txt
Sudo fdisk –l >before.txt
Diff after.txt before.txt
We can find it mount on /dev/sdb1

Problem2

Hostname 查看主机名字 Hostnamectl 可用于查询和更改系统主机名和相关设置




```
sudo hostnamectl set-hostname linuxconfig
```

修改文件/etc/hosts，这样重启也依旧是更改后的名字

Problem3

Ssh 连接服务器，我把 demo_x.c 传输上去. 然后修改成雨伞或者其他图标,display 一下就可以了.

scp -P 11189 -r demo_x.c root@117.93.33.92:/home/data

不用登录 ssh 也可以用 scp 把 demo_x.c 传输到 117.93.33.92 的/home/data/文件夹下,用户名为 root

```
[root@yishangyun home]# cd ..
[root@yishangyun /]# cd usr
[root@yishangyun usr]# ls
bin  etc  games  include  lib  lib64  libexec
[root@yishangyun usr]# cd ..
[root@yishangyun /]# ls
bin  boot  dev  etc  home  lib  lib64  media
[root@yishangyun /]# cd home
[root@yishangyun home]# cd data
[root@yishangyun data]# ls
demo_x.c
[root@yishangyun data]#
```

gcc -o main demo_x.c -lX11 编译

修改服务器和客户端的 ssh 配置文件

```
xhost +     //允许服务器的的 x11 界面连接过来
ssh -P 11189 -X root@117.93.33.92          //-X 参数表示转发 X11 数据
```

然后就可以运行程序

```
xclock
```

```
luke@linuxconfig:~$ sudo vim /etc/ssh/ssh_config
[sudo] luke 的密码:
linjuyi对不起，请重试。
[sudo] luke 的密码:
luke@linuxconfig:~$ sudo systemctl restart ssh.ser
luke@linuxconfig:~$ xhost +
access control disabled, clients can connect from
luke@linuxconfig:~$ ssh -X -p 11189  root@117.93.3
root@117.93.33.92's password:
Last login: Mon Jan 18 13:45:58 2021 from 183.157.
[root@yishangyun ~]# xclokc
bash: xclokc: 未找到命令...
相似命令是：  'xclock'
[root@yishangyun ~]# xclock
Warning: Missing charsets in String to FontSet con
xclock...
```

修改代码如下,即可显示雨伞

```
hints.min_width  = 16; hints.min_height = 16;
hints.width_inc = hints.height_inc = 16;
hints.flags |= PMaxSize | PMinSize | PResizeInc;
//xswa.cursor = XCreateFontCursor (theDisp, XC_box_spiral);
xswa.cursor = XCreateFontCursor (theDisp, XC_umbrella);
xswamask = CWCursor;
```

Problem4

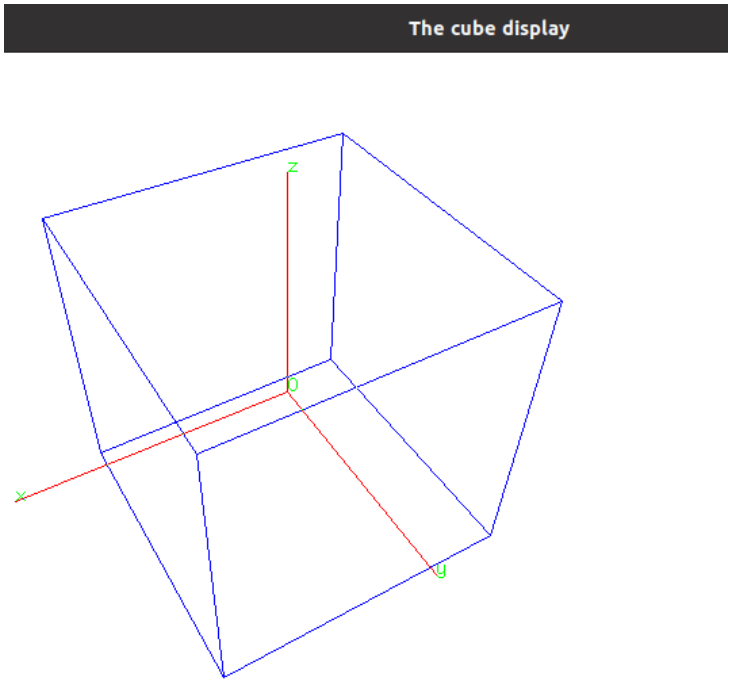
```
luke@linuxconfig:~/tmp$ make
gcc -g -c demo_bit.c
as -o demo_bit_count.o demo_bit_count.s
***demo_bit.o demo_bit_count.o is(are) newer than demo_bit
gcc -g -o demo_bit demo_bit.o demo_bit_count.o
```

the purpose of the result program. : turn the input into hexadecimal number and count how many bits of this number is 1;

problem5

make and display

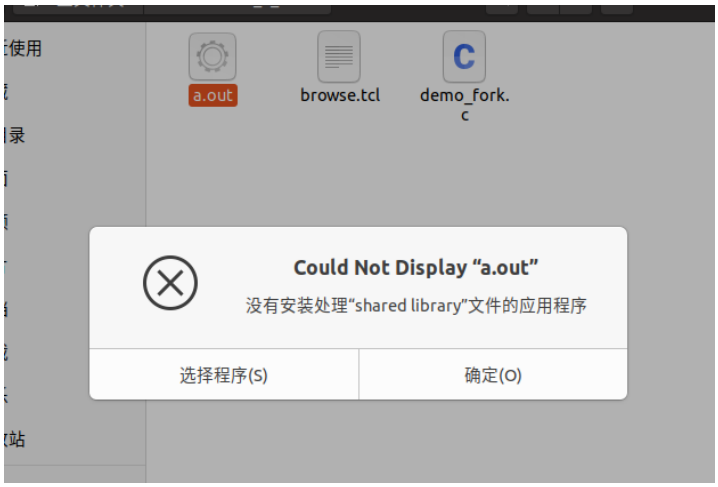
```
[root@yishangyun data]# cd cube_wrong/
[root@yishangyun cube_wrong]# make
gcc -g -c -I /usr/include/X11 -I ./include main.c -o main.o
gcc -g -c -I /usr/include/X11 -I ./include init.c -o init.o
gcc -g -c -I /usr/include/X11 -I ./include readline.c -o readline.o
gcc -g -c -I /usr/include/X11 -I ./include toview.c -o toview.o
gcc -g -c -I /usr/include/X11 -I ./include translate.c -o translate.o
gcc -g -c -I /usr/include/X11 -I ./include cast.c -o cast.o
gcc -g -c -I /usr/include/X11 -I ./include drawline.c -o drawline.o
gcc -g -c -I /usr/include/X11 -I ./include draw_guide.c -o draw_guide.o
gcc -g main.o init.o readline.o toview.o translate.o cast.o drawline.o draw_guid
e.o -L /usr/lib/x86_64-linux-gnu -lX11 -lm -o draw
[root@yishangyun cube_wrong]#
```



the reasons for compiling warnings
1. don't have the header file
2. don't have correct number of arguments

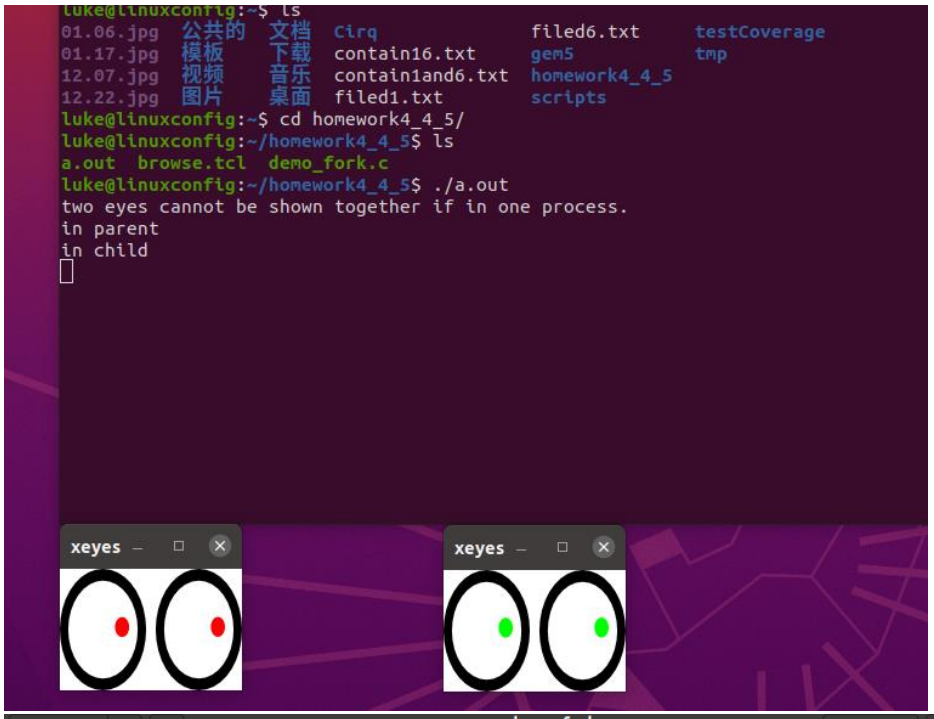
Problem 1. Build 'demo_fork.c' and run it, then try to change the red eyes to 'browse.tcl' and run it again.

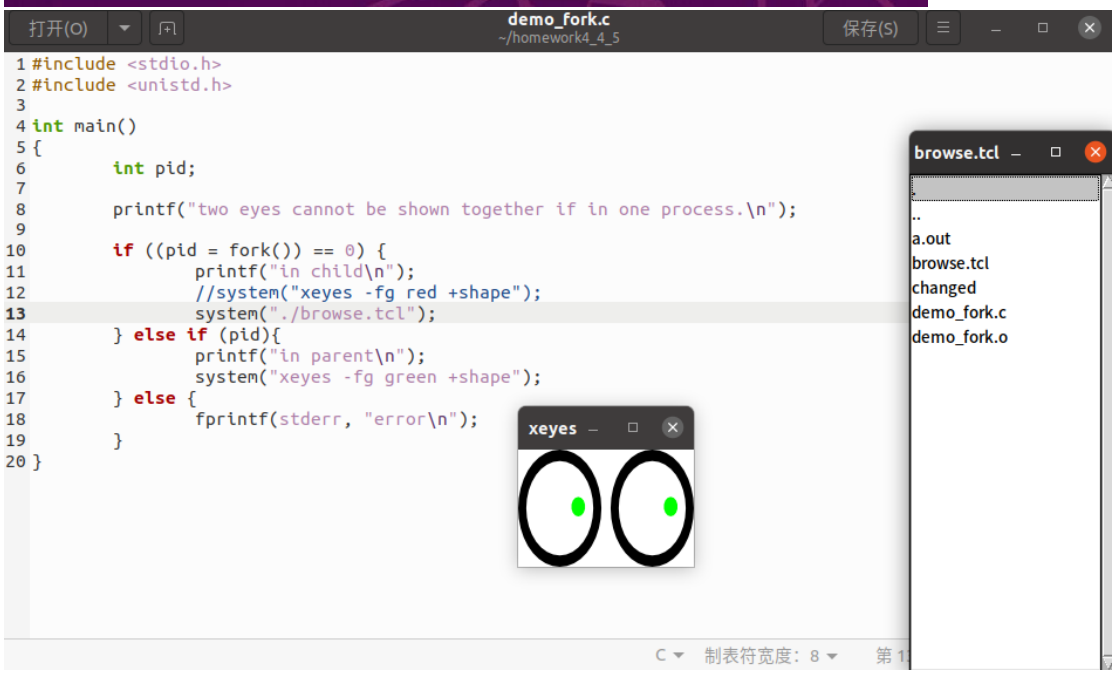
通过 C 代码启动 tcl?
错误 1 Gcc demo_fork.c
不能执行 a.out



解决方法:

终端中打开,成功





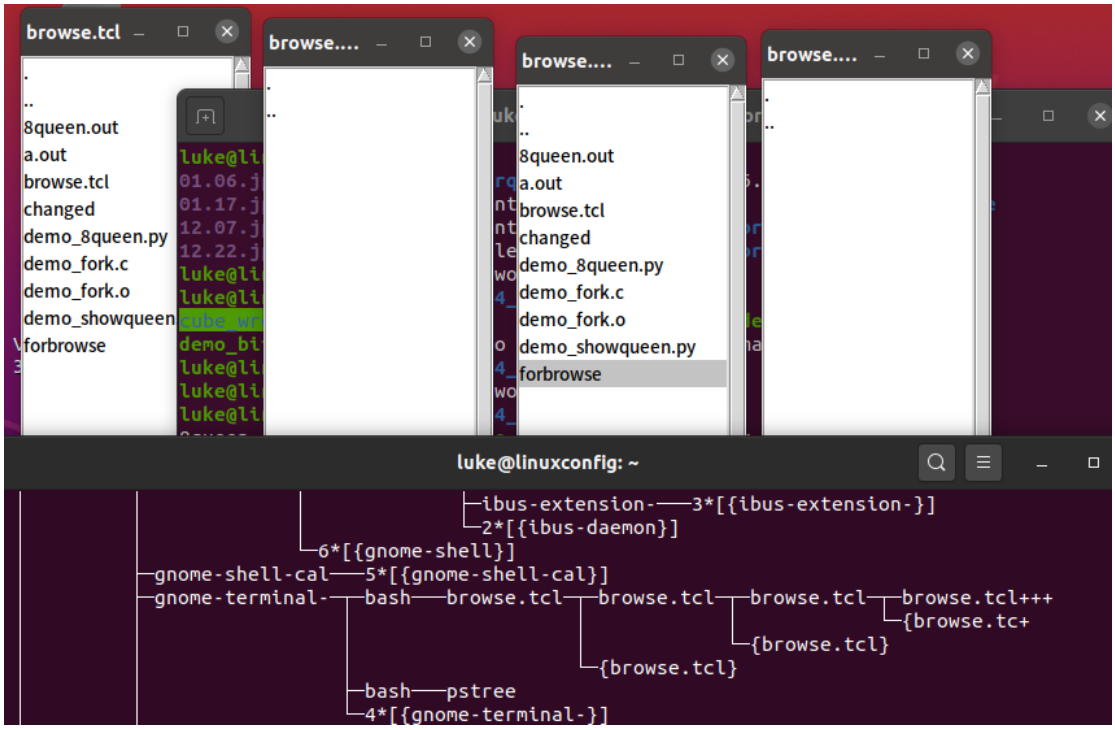
成功把红眼睛改为 browse.tcl'

错误 2 : bash: ./browse.tcl: /usr/bin/wish: 解释器错误: 没有那个文件或目录

解决方法: `apt-get install -y wish ,就可以了.`

Invoke more directory browsers from browse.tcl. Use 'pstree' to see the process hierarchy for all of them,

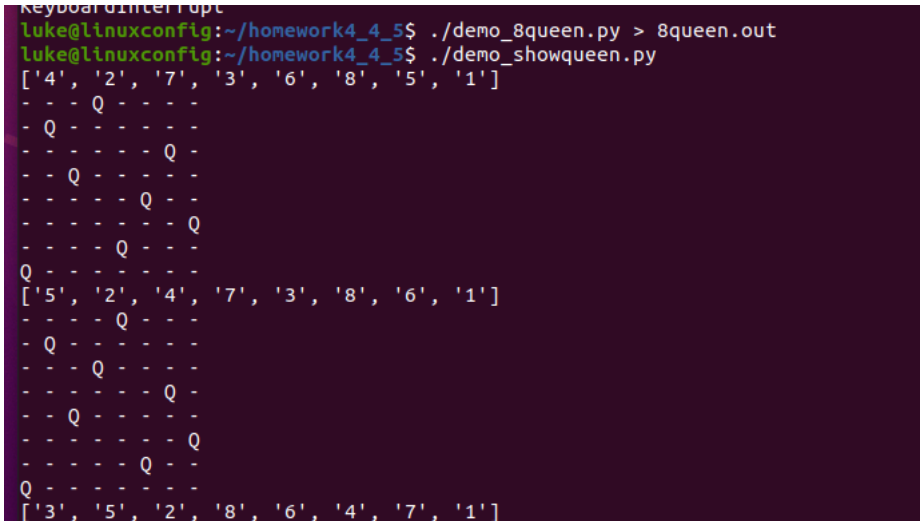
再点击一个目录就可以了 and show the hierarchy with a screenshot.如下图



Problem 2. Run Python scripts of demo_8queen.py and demo_showqueens.py, read the source code, comments and their outputs.

```
luke@linuxconfig:~/homework4_4_5$ ./demo_8queen.py
[(1, 4), (2, 2), (3, 7), (4, 3), (5, 6), (6, 8), (7, 5), (8, 1)]
[(1, 5), (2, 2), (3, 4), (4, 7), (5, 3), (6, 8), (7, 6), (8, 1)]
[(1, 3), (2, 5), (3, 2), (4, 8), (5, 6), (6, 4), (7, 7), (8, 1)]
[(1, 3), (2, 6), (3, 4), (4, 2), (5, 8), (6, 5), (7, 7), (8, 1)]
[(1, 5), (2, 7), (3, 1), (4, 3), (5, 8), (6, 6), (7, 4), (8, 2)]
[(1, 4), (2, 6), (3, 8), (4, 3), (5, 1), (6, 7), (7, 5), (8, 2)]
[(1, 3), (2, 6), (3, 8), (4, 1), (5, 4), (6, 7), (7, 5), (8, 2)]
[(1, 5), (2, 3), (3, 8), (4, 4), (5, 7), (6, 1), (7, 6), (8, 2)]
[(1, 5), (2, 7), (3, 4), (4, 1), (5, 3), (6, 8), (7, 6), (8, 2)]
[(1, 4), (2, 1), (3, 5), (4, 8), (5, 6), (6, 3), (7, 7), (8, 2)]
[(1, 3), (2, 6), (3, 4), (4, 1), (5, 8), (6, 5), (7, 7), (8, 2)]
[(1, 4), (2, 7), (3, 5), (4, 3), (5, 1), (6, 6), (7, 8), (8, 2)]
[(1, 6), (2, 4), (3, 2), (4, 8), (5, 5), (6, 7), (7, 1), (8, 3)]
[(1, 6), (2, 4), (3, 7), (4, 1), (5, 8), (6, 2), (7, 5), (8, 3)]
[(1, 1), (2, 7), (3, 4), (4, 6), (5, 8), (6, 2), (7, 5), (8, 3)]
[(1, 6), (2, 8), (3, 2), (4, 4), (5, 1), (6, 7), (7, 5), (8, 3)]
[(1, 6), (2, 2), (3, 7), (4, 1), (5, 4), (6, 8), (7, 5), (8, 3)]
[(1, 4), (2, 7), (3, 1), (4, 8), (5, 5), (6, 2), (7, 6), (8, 3)]
[(1, 5), (2, 8), (3, 4), (4, 1), (5, 7), (6, 2), (7, 6), (8, 3)]
```

输出到 8queen.out 即可



Then (OPTIONALLY) write a Perl or Python script to analyze the solutions in 8queen.out, and show how many kinds of the solutions are unique.

(symmetrical solutions are not considered as unique, for example, one solution being mirrored or rotated should be eliminated from the unique list)